

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problems Mailbox.**

This Page Blank (uspto)



(19)

Europäisches Patentamt

European Patent Office

## Office européen des brevets



(11)

EP 0 987 627 A1

(12)

## EUROPEAN PATENT APPLICATION

(43) Date of publication:  
22.03.2000 Bulletin 2000/12

(51) Int. Cl.<sup>7</sup>: **G06F 9/445, G06F 9/44**

(21) Application number: 98117504.5

(22) Date of filing: 15.09.1998

(84) Designated Contracting States:  
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU  
MC NL PT SE  
Designated Extension States:  
AL LT LV MK RO SI

(71) Applicant: EASTMAN KODAK COMPANY  
Rochester, New York 14650 (US)

(72) Inventor:  
**Jooss, Frank**  
**c/o Kodak Aktiengesellschaft**  
**70323 Stuttgart (DE)**

(74) Representative:  
Reichert, Werner F. Dr. Dipl.-Phys.  
Kodak Aktiengesellschaft,  
Patente, 1303  
70323 Stuttgart (DE)

**Remarks:**

Claim 18 is deemed to be abandoned due to non-payment of the claims fee (Rule 31 (2) EPC).

**(54) Method and system for performing an update**

(57) A method and a system for performing an update of a plurality of external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) are disclosed. The system comprises a central (22) computer, which is connected by a communication link to at least one external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>). A structured memory in the central computer (22) has a subdirectory (30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub>) for each of the external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>). Each subdirectory (30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub>) is further partitioned into at least one further subdirectory (31<sub>1</sub>, 31<sub>2</sub>, 31<sub>3</sub>, ...31<sub>n</sub>; 32<sub>1</sub>, 32<sub>2</sub>, 32<sub>3</sub>, ...32<sub>n</sub>; 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub>) for holding files to be updated. Each of the external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) comprise a computer with a memory. The memory is structured for holding the same subdirectories (34<sub>1</sub>, 34<sub>2</sub>, 34<sub>3</sub>, ...34<sub>n</sub>; 35<sub>1</sub>, 35<sub>2</sub>, 35<sub>3</sub>, ...35<sub>n</sub>; 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub>) as implemented in the central computer (22).

**Description****FIELD OF THE INVENTION**

5 [0001] The invention relates to a method for performing an update. More particularly the method performs an update of a plurality of external stations.

[0002] Additionally, the invention relates to a system for performing an update. The system comprises a central computer, at least one external station, a communication link between said at least one external station and said central computer for transferring files from said central computer to at least one external station.

10

**BACKGROUND OF THE INVENTION**

[0003] There are numerous suggestions in the art which are used to carry out an update of a remote electronic device. The files used for the update are located in the memory of a central computer or host. Documents disclosing related prior art listed below:

[0004] The document WO 98/21865 discloses an apparatus and a method for optimizing the use of long distance calls. A common telephone is connected to an additional device storing the rates of various telecommunication companies. In case a user places a certain long distance call the additional device retrieves the company which offers the best rate for the particular phone call. The additional device is connected to a central computer via the usual phone line. The computer regularly updates the additional device with respect to the rates and the time the best rates are offered. The data are stored in an internal memory of the device.

[0005] The document EP-A-0 836 138 provides a method, using a management information system, for installing and updating software on several computers. The management information system establishes a connection to the remote computers and checks if there is a need for an update or not. In case an update is necessary the management information system checks if the connection is suitable for the download and update of the software. If the result of the test is positive the management information system carries out the download and the update of the software on the remote computers. The connection between the management information system and the remote computers can be a local area network or a common telephone line.

[0006] A further method for updating computer software is disclosed in EP-A-0 841 615. A computer has a memory which stores several software applications. Additionally, a file is present which indicates the actual software version installed in the memory. An update mechanism installed in the computer, accesses a central server in order to import updates from there. The update mechanism looks up the newest update version available on the central server. A file, also available on the central server, comprises further details necessary for carrying out the updates. This file controls the transfer of files from the central server to the computer such that only those files are transferred to the computer which do not conform with the newest version.

15

20

25

30

35

**SUMMARY OF THE INVENTION**

[0007] The present invention provides a method for performing an update of a plurality of external stations. Especially, the external stations are order stations which are established at a retailer store location. An order station can be used by a customer to create a personal order which is later processed in a remote location. For example, a customer order processed at the remote location is the creation of prints out of a set of digital image data.

[0008] It is the object of the invention to provide a method which provides a simple, secure and inexpensive way to update each of the external stations.

45

[0009] The above object is accomplished by the steps of:

- writing at least one file, to be used for update into a subdirectory of a memory of a central computer, wherein for each external station a specific subdirectory is implemented;
- establishing connection with the central computer and the external stations;
- transferring the at least one update file to each of said external stations which are in communication with the central computer;
- checking the size of the at least one file transferred;
- storing the at least one file in a memory of each of the external stations which are in communication with the central computer;
- carrying out the update in each of the external stations which have properly received the transferred files for update; and
- writing an update report in the memory of each of said external stations.

[0010] It is a further object of the present invention to provide a system which enables a centrally controlled and monitored update of a plurality of external stations.

[0011] The above object is accomplished by a system characterized by:

- 5 • a structured memory in the central computer holding a subdirectory for each of the external stations, said subdirectory has at least one further subdirectory for holding files to be updated, and
- a computer in each of the external stations having a memory for holding the same said at least one further subdirectory for receiving said files to be updated.

10 [0012] The advantages of the inventive method and system are that the update of a plurality of order stations can be carried out without the need of service personnel to show up at each location of the plurality of external stations. Additionally, the update of the external stations will be effective in all active external service stations at the same time and date. This is especially important if the update effects a price change of a customer order or a part of the latter. The redundant subdirectory structure in the central computer and in each of the external stations opens up the possibility to monitor the update. Furthermore, it enables the update of external stations, which have not been "ON" or ready for the update at a specific time or date the update is preferably been carried out. Further advantages or embodiments of the invention become apparent from the dependent claims.

#### BRIEF DESCRIPTION OF THE DRAWINGS

20 [0013] The subject matter of the invention is described with reference to the embodiment shown in the drawings.

Fig. 1 is a perspective view of an order station, which is an embodiment of an external station to be updated with the system of the present invention;  
 25 Fig. 2 is a schematic representation of the installation of the inventive system;  
 Fig. 3 displays a flow chart of the update process;  
 Fig. 4 displays a flow chart of the data update process;  
 Fig. 5 displays a flow chart of the software update process; and  
 Fig. 6 displays a flow chart of the infile update process.

#### DETAILED DESCRIPTION OF THE INVENTION

[0014] The invention can be used to update any type of external station from a single central computer. The connection or communication link between the plurality of external stations and the central computer is established via a conventional telephone line or via internet. The software for establishing the connection and the connection data for each external station are stored in a memory of the central computer. Preferably the invention is applied to update data, software and files of an order station or a plurality of order stations which are in operation at different locations. The construction and the functionality of the order station is mentioned briefly in the following paragraph. A more detailed description is disclosed in the German Patent Applications DE-A-198 02 026.0 and DE-A-198 02 017.1.

[0015] The order station 2, as disclosed in Fig. 1, comprises a housing 4 which is open at one side. Inside the housing 4 several devices are provided; for example a monitor 10, a computer 12, an optional storage compartment 14 for digital cameras to which several cable guides 15 are connected for cables for connecting digital cameras, and a document printer 16. The housing is closed off to the outside by a front panel 6, the latter having, in the area of the monitor, a glass sheet with touch recognition.

[0016] The placement of an order of ordering prints of digitized picture files at an order station 2 is as follows:

selecting on a starting interface one language that is used during the ordering process, touching an active field assigned to the required language on the glass sheet and the language is loaded from the computer 12 and displaying the language accordingly on the monitor 10;

50 selecting a carrier for the digitized picture files displayed by the computer 12 on an input medium interface on the monitor 10, displaying an index interface on the monitor 10 by the computer 12 after touching an active field assigned to said required carrier on the glass sheet 18;

selecting at least one picture to be included in the order out of the pictures displayed on said index interface, displaying a format selection interface on the monitor 10 by the computer 12 after touching an active field assigned to the required picture on the glass sheet;

55 selecting a format in said format selection interface by touching an active field assigned to a preset format on the glass sheet 18, and confirming the selection by touching a confirmation icon to which an active field has been assigned on the glass sheet 18, and displaying a number selection interface on the monitor 10 by the computer 12;

inputting the number of prints for the selected picture in said number selection interface by touching active fields assigned to a key field on the glass sheet, and confirming the input by touching a confirmation icon to which an active field has been assigned on the glass sheet 18, and displaying an order overview interface on the monitor 10 by the computer 12;

5 confirmation icon to which an active field has been assigned on the glass sheet 18, and thereby displaying an address input interface on the monitor 10 by the computer 12;

inputting the user data in said address input interface by touching active fields that are assigned to a key field on the glass sheet, and confirming the input by touching a confirmation icon to which an active field has been assigned 10 on the glass sheet 18, and thereby displaying a confirmation interface on the monitor 10 by the computer 12; and confirming the order in said confirmation interface by touching a confirmation icon to which an active field has been assigned on the glass sheet 18, and thereby printing a receipt by the computer 12 using the document printer and then transmitting the order to a central printing station via a data line.

[0017] The monitor 10 of the order station 2 displays information, which is placed in separate windows or fields. Some 15 of the information fields may be overlaid by a touch recognition to allow a user to place a customized order and to guide him through the complete ordering process. There is a certain need to provide the possibility to easily update or alter the information, the data, the fields itself or the software.

[0018] Fig. 2 shows schematically a setup of a system for updating the at least one order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. The description of the update of at least one order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is not regarded as a limitation, since a 20 person skilled in the art would apply the teaching to other external stations which perform different functions as the specific order station. The automatic update is monitored by a central computer 22. The central computer 22 is connected via a conventional telephone line 28 to each of the order stations 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. The central computer 22 has a structured memory (not shown). For each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> a subdirectory 30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub> is provided which stores the data or the files needed for update of each of the order stations 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. Each of the 25 subdirectories 30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub> is partitioned into at least a first 31<sub>1</sub>, 31<sub>2</sub>, 31<sub>3</sub>, ...31<sub>n</sub>, a second 32<sub>1</sub>, 32<sub>2</sub>, 32<sub>3</sub>, ...32<sub>n</sub> and a third 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> subdirectory. The first subdirectory 31<sub>1</sub>, 31<sub>2</sub>, 31<sub>3</sub>, ...31<sub>n</sub> is used to store information about the customer orders which have been carried out at the respective order station. The second subdirectory 32<sub>1</sub>, 32<sub>2</sub>, 32<sub>3</sub>, ...32<sub>n</sub> is used to store information which provides the retailer or the service provider with statistic data. The 30 data of interest are: how long does it take for a customer to place an order, what data sources are used, how often the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> was used and no order has been sent off, etc. The third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> stores all the files and data used to carry out an update of the order stations 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. The same subdirectory structure as for each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> on the central computer 22 is present in the memory of each computer 12 of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. As mentioned above with respect to the central computer 22 a subdirectory 26<sub>1</sub>, 26<sub>2</sub>, 26<sub>3</sub>, ...26<sub>n</sub> of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is partitioned into at least a first 34<sub>1</sub>, 34<sub>2</sub>, 34<sub>3</sub>, ...34<sub>n</sub>, a second 35<sub>1</sub>, 35<sub>2</sub>, 35<sub>3</sub>, ...35<sub>n</sub> and a third 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> subdirectory. The content of the subdirectories implemented in each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is the same as the content of the subdirectories 30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub> in the memory of the central computer 22.

[0019] The update process of the external station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is described in Fig. 3. As mentioned above, the 40 following description uses the example of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> (see Fig. 1) to make the update process more clear to the reader. Prior update of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> the subdirectory 30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub> for each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> has to be provided in the memory of the central computer 22. The data or files required for the update of each individual order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> stored in each subdirectory 30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub> at the appropriate place. As mentioned above the files and data for the update are stored in the third 45 subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22. It is preferred to carry out the update of the order stations 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> at a time at which it is unlikely that the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is busy. A possible time for carrying out the update would be 4:00 a.m. This assumes that the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is still ON. At the time mentioned above the central computer 22 tries to establish the connection with each of the order stations 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> via a conventional telephone line 28 or internet. The connection is successfully established with those order stations 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> which are ON. A connection between the central computer 22 and those order stations, which 50 were not ON at the specific time the connection to the central computer was established, is generated as soon as the order station is switched ON.

[0020] For the following description we assume that the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is ON and the connection with the central computer 22 is successfully established. Prior to the transfer of the files from the central computer 22 to each of the individual order stations 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>, a check is made by each order station if an update is necessary. The update process of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is stopped if the software or file version is the same 55 as the update version provided by the central computer 22. In case an update is due, a copy of the files necessary for the update is transferred from each third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22 to the related third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of the specific order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. The files are stored in memory of each

order station  $20_1, 20_2, 20_3, \dots, 20_n$  having the same subdirectory structure as in the central computer 22. The files transferred from the central computer 22 are checked with respect to file size in order to detect any wrong transmission or wrong data. In case any difference in file size is detected, the transfer of the file to the specific order station  $20_1, 20_2, 20_3, \dots, 20_n$  has to be repeated.

5 [0021] After the successful transfer of the file, the update process is carried out and the data representing the date, the software version, etc. are stored in a file in the memory of each order station  $20_1, 20_2, 20_3, \dots, 20_n$ . Additionally, each order station  $20_1, 20_2, 20_3, \dots, 20_n$  sends a response to the central computer 22. The response is attached to a file in the memory of the central computer 22. The file is sent to and stored in each third subdirectory  $33_1, 33_2, 33_3, \dots, 33_n$  of the central computer 22 related to each order station  $20_1, 20_2, 20_3, \dots, 20_n$ .

10 [0022] Three different updates are carried out simultaneously or one by one on each of the order stations  $20_1, 20_2, 20_3, \dots, 20_n$ . The first update is a "DATA" update (see Fig. 4) which is carried out on a specific date. This update is substantially used to update the prices or the formats of the prints, which are sent off by a user of an order station  $20_1, 20_2, 20_3, \dots, 20_n$ . The central computer 22 holds in the third subdirectory  $33_1, 33_2, 33_3, \dots, 33_n$  a file, for example named "dataupdate.ini", which is structured as follows:

15 [DATA]

[0023]

20 Date = (the date for carrying out the update) For example: 20.12.1998  
 Filename = (filename of the file to be replaced) For example: Date.ini  
 Checksum = (size of the file to be replaced in Bytes)

25 [0024] As mentioned above, "Date" is the date on which the change or the update should be effective. The date has to be written in the form "dd.mm.yyyy", wherein "dd" stands for the day, "mm" for the month and "yyyy" for the current year. The file to be replaced, here "Date.ini" has to be in the memory of the central computer 22 in the third subdirectory  $33_1, 33_2, 33_3, \dots, 33_n$  for each order station  $20_1, 20_2, 20_3, \dots, 20_n$ . The file has to be present in the specific third subdirectory  $33_1, 33_2, 33_3, \dots, 33_n$  until the update is effective in the specific order station  $20_1, 20_2, 20_3, \dots, 20_n$ . "Checksum" represents the number of Bytes of the file, transferred from the central computer 22 to the order station  $20_1, 20_2, 20_3, \dots, 20_n$ .  
 30 It is checked if the size of the file stored in the central computer 22 differs from the file transferred to the order station  $20_1, 20_2, 20_3, \dots, 20_n$ . In case of any difference, the transfer has to be carried out once more. In case an order station  $20_1, 20_2, 20_3, \dots, 20_n$  is not ON at the specific date, the update has to be carried out as soon as the order station  $20_1, 20_2, 20_3, \dots, 20_n$  is ON. A comparison is made if the present date is greater than the date of the update. If it is true, the update is carried out even after the due date of the update.

35 [0025] Additionally, there is a possibility that an order station  $20_1, 20_2, 20_3, \dots, 20_n$  is switched ON and OFF several times a day. In order to avoid the same update more than one time, the order station  $20_1, 20_2, 20_3, \dots, 20_n$  stores after a successful update the date in a file, for example, "updateVersion.ini". Prior to the start of an update, a comparison is made between the date stored in "update.ini" in the third subdirectory  $33_1, 33_2, 33_3, \dots, 33_n$  of the specific order station on the central computer 22 and the date stored in "updateVersion.ini" in the respective subdirectory  $36_1, 36_2, 36_3, \dots, 36_n$  on the order station  $20_1, 20_2, 20_3, \dots, 20_n$ . If a match of the dates stored is detected, no update needs to be done since the update has already been carried out.

40 [0026] This update does not request a restart of the order station  $20_1, 20_2, 20_3, \dots, 20_n$ . Therefore, all files may be substituted which are not constantly used. Files with the extension ".DLL" or ".EXE" cannot be updated with this type of update. This update procedure is suitable for files, which are opened and used several times during runtime of the order station  $20_1, 20_2, 20_3, \dots, 20_n$ .

45 [0027] The second update is a "SOFTWARE" update (see Fig. 5) which is carried out if there is a difference in the software versions used on the order stations  $20_1, 20_2, 20_3, \dots, 20_n$ , and a new version stored in each specific third subdirectory  $33_1, 33_2, 33_3, \dots, 33_n$  of the central computer 22. The central computer 22 holds in the third subdirectory a file, for example, named "softwareupdate.ini", which is structured as follows:

50 [SOFTWARE]

[0028]

55 Version = (Version of the update)  
 File1 = (filename of the first file to be replaced)  
 File2 = (filename of the second file to be replaced)  
 File3 = (filename of the third file to be replaced)

Checksum = (size in Bytes of all files to be replaced)

[0029] Data about the version of the update is necessary to identify and select the correct update. The version number is represented by a string of characters and stored, for example, in the file "updateVersion.ini", in the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>.

[0030] At 4:00 a.m. or during switch "ON" of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> the file "update.ini" is read from the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22 and the version number is compared with the version number stored in the file "updateVersion.ini" in the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub>, of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. If an update is required, the files present in the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22 are transferred to and copied into the corresponding third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub>, of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. Additionally, each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> and the central computer 22 carry out a check to determine a difference between the file size on the central computer 22 and the file size received by each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. In case any mismatch is detected the transfer of the file or files is repeated. Then a check is made to detect if the order station is currently processing an order or is busy. The order station marks in a file "Messages.ini" the status of order station, wherein the "busy status" is marked with 0 and the "ready status" is marked with 1. If the order station is busy, the carrying out of the update is delayed one minute. After one minute the status of the order station is checked again. As soon as the order station has reached the "ready" status, the operation program "ROS.EXE" of the order station is stopped (please note the program responsible for the transfer, "Transfer.exe", of files between the central computer 22 and the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is not terminated). The files, which are present in the third specific subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22, are copied to the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of the order station and the operation program "ROS.EXE" is restarted.

[0031] The restart of "ROS.EXE" causes an automatic restart of "Transfer.exe". Therefore, "Transfer.exe" has to be terminated prior to the restart of the entire system. Additionally, the file "updateVersion.ini" has to be written which contains the number of the newest software version installed on each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>.

[0032] Since Windwos95™ does not allow the replacement of a file as long as it is in use, "Transfer.exe" cannot be updated the way as described above. "Transfer.exe" cannot be copied form the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22 to the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. The copy process of "Transfer.exe" is done by "ROS.EXE". Prior to the start of "Transfer.exe", initiated by the start of "ROS.EXE", it is checked if "transfer.exe" is in the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22. If this is true, the system controls the activity of "Transfer.exe" several times, until a termination of the program "Transfer.exe" is detected. Upon this result the new version of the file "Transfer.exe" is copied into the memory and restarted.

[0033] The third update is a "ROS" update (see Fig. 6), which is carried out if there is a need to change the entries in specific inifiles effective in each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. To change the entries a file "iniupdate.ini" is used, which the central computer 22 holds in the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub>. The structure of the file is as follows:

[VERSION]

[0034]

40 Version = (Version of the update) The number of the version can be similar or different to the number of the version used for updating the software of the order stations.

[INI]

45. Filename = (name of the inifile in which the changes have to be made)

[0035]

Section1 = (first section ([ ]) of the inifile)	For example: SECTION1
Key1 = (first key)	For example: KEY1
String1 = (text to be replaced)	For example: STRING1.
Section2 (second section ([ ]) of the inifile)	For example: SECTION 2
Key2 = (second key)	For example: KEY2
String2 = (text to be replaced)	For example: STRING2

55 [0036] Additionally, information about the version of the update is necessary to identify and select the correct update. The version number is represented by a string of characters and stored for example in the file "updateVersion.ini" in the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>.

[0037] At 4:00 a.m. or during switch "ON" of the order station, the file "update.ini" is read from the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22 and the version number is compared with the version number stored in the file "updateVersion.ini" in the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. If an update is required, the files present in the third subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22 are transferred to and copied into the corresponding third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. Additionally, each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> and the central computer 22 carry out a check to determine a difference between the file size on the central computer 22 and the file size received by each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>. In case any mismatch is detected the transfer of the file or files is repeated. A check is made to detect if the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is currently processing an order or is busy. The order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> marks in a file "Messages.ini" the status of order station, wherein the "busy status" is marked with 0 and the "ready status" is marked with 1. If the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is busy, the carrying out of the update is delayed one minute. After one minute the status of the order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> is checked again. As soon as the order station has reached the "ready" status the operation program "ROS.EXE" of the order station is stopped. The files, which are present in the third specific subdirectory 33<sub>1</sub>, 33<sub>2</sub>, 33<sub>3</sub>, ...33<sub>n</sub> of the central computer 22, are copied to the third subdirectory 36<sub>1</sub>, 36<sub>2</sub>, 36<sub>3</sub>, ...36<sub>n</sub> of the order station and the operation program "ROS.EXE" is restarted. The restart of the program "ROS.EXE" makes the amendment in the inifiles effective.

[0038] The file "updateVersion.ini" has to be written which contains the date of the newest update of the inifiles installed on each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>.

[0039] During every update, regardless of the type of the update, an additional program "PIROSupdate.exe" present on the central computer 22, protocols for each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> all the actions which have been carried out while the update was in progress. The data are stored in a file "StationXXXXXX.txt". The X's represent the individual number of each order station 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub> placed at a retail store location. Every new entry is attached to the end of the file. A block of data may look as follows:

25 \*\* Data Update \*\*

[0040]

30 Date = ( Date when Data Update has been carried out)  
Filename = ( Name of the file to be replaced)

\*\* Software Update \*\*

[0041]

35 Version = ( Version Number of Software Update)  
Filename1 = (First file to be replaced)  
Filename2 = (Second file to be replaced)

40 \*\* INI Update \*\*

[0042]

45 Version = ( Version Number of INI Update)  
Infilename = (file name of the Infile)  
Section1 = ( Section1 to be amended)  
Key1 = ( Key1 to be amended)  
String1 = ( Text to be replaced)

50 [0043] A file "UpdateProtocol.txt" is located in the third subdirectory of the memory of the central computer 22. After every update a report is attached to the end of this file. The data have for example the following form:

[DATA UPDATE]

55 [0044]

date = ( Date of Data Update)

[SOFTWARE UPDATE]

[0045]

5 Version = (Version Number of Software Update)

[INI UPDATE]

[0046]

10 Version = (Version Number of INI Update)

[UPDATE STATUS]

15 [0047]

Date = (implementation date)

Time = (time of implementation)

Status = "Update successful"

20 [0048] The data above show the type of update (written in brackets) and the number of the actual version or the date. It is clear from the description of the update process that it is not necessary to carry out all three types of updates at the same time. In case the update was successful, the above last four lines are attached to the end of the file. Date represents the date on which the update has been carried out. Time represents the time on which the update has been carried out. The last line finally shows that the update was successful. No update was successfully carried out if the last line of the above file is not present.

25 [0049] The invention has been described in detail with particular reference to certain preferred embodiments thereof, but it will be understood that variations and modifications can be effected within the scope of the invention.

30 **Claims**1. A method for performing an update of a plurality of external stations 20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) characterized by the steps of:35

- writing at least one file, to be used for update into a subdirectory of a memory of a central computer (22) wherein for each external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) a specific subdirectory (30<sub>1</sub>, 30<sub>2</sub>, 30<sub>3</sub>, ...30<sub>n</sub>) is implemented;
- establishing connection with the central computer (22) and the external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>);
- transferring the at least one update file to each of said external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) which are in communication with the central computer (22);
- checking the size of the at least one file transferred;
- storing the at least one file in a memory of each of the external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) which are in communication with the central computer (22);
- carrying out the update in each of the external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) which have properly received the transferred files for update; and
- writing an update report in the memory of each of said external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).

40 2. The method as claimed in claim 1 wherein the at least one transferred file is a file for updating data on said external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).45 3. The method as claimed in claim 1 wherein the at least one transferred file is a file for updating software on said external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).50 4. The method as claimed in claim 1 wherein the at least one transferred file is a file for updating an ini file on said external stations (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).

55 5. The method as claimed in claim 2 comprises the additional steps of:

- establishing connection with the central computer (22) and the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) at a specific time and on a specific date;
- comparing the present date with the date of the update and transferring upon comparison the file to be updated to each of the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ); and
- storing the date of the update in a file on the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ).

5        6. The method as claimed in claim 3 comprises the additional steps of:

- establishing connection with the central computer (22) and the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) at a specific time;
- comparing the present version of files to be updated with the version of the update and transferring upon comparison the files to be updated to each of the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ );
- storing the date of the update in a file on the external station; and
- restarting the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ).

10        7. The method as claimed in claim 4 comprises the additional steps of:

- establishing connection with the central computer (22) and the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) at a specific time;
- comparing the present version of the inifiles to be updated with the version of the inifiles presented for update and transferring upon comparison the inifiles to be updated to each of the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ );
- storing the date of the update in a file on the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ); and
- restarting the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ).

15        8. The method as claimed in claim 1, 5, 6 or 7 wherein the step of establishing connection with those external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) is deferred until the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) are ON.

20        9. The method as claimed in claim 1, 5, 6 or 7 wherein the step of transferring the file to be updated from the central computer (22) to the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ) is carried out again if there is any mismatch in file size detected.

25        10. The method as claimed in claim 6 or 7 wherein the step of carrying out the update on those external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) is deferred until the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) have left the busy state.

30        11. A system for performing an update comprises a central (22) computer, at least one external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ), a communication link between said at least one external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ) and said central computer (22) for transferring files from said central computer (22) to said at least one external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ); the system is characterized by:

- a structured memory in the central computer (22) holding a subdirectory ( $30_1, 30_2, 30_3, \dots, 30_n$ ) for each of the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) said subdirectory ( $30_1, 30_2, 30_3, \dots, 30_n$ ) has at least one further subdirectory ( $31_1, 31_2, 31_3, \dots, 31_n, 32_1, 32_2, 32_3, \dots, 32_n, 33_1, 33_2, 33_3, \dots, 33_n$ ) for holding files to be updated, and
- a computer in each of the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ) having a memory for holding the same said at least one further subdirectory ( $34_1, 34_2, 34_3, \dots, 34_n, 35_1, 35_2, 35_3, \dots, 35_n, 36_1, 36_2, 36_3, \dots, 36_n$ ) for receiving said files to be updated.

35        12. The system as claimed in claim 11 wherein the update files comprise a first file type for enabling a data update in the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ), a second file type for enabling a software update in the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ) and a third file type for enabling an update of inifiles in the external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ).

40        13. The system as claimed in claim 11 wherein the communication link between the central computer (22) and the at least one external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ) is established via a conventional telephone line and the internet.

45        14. The system as claimed in claim 11 wherein three subdirectories are established on the central computer (22) for each external station ( $20_1, 20_2, 20_3, \dots, 20_n$ ) and the same three subdirectories are established in each of the external stations ( $20_1, 20_2, 20_3, \dots, 20_n$ ).

15. The system as claimed in claim 14 wherein a first subdirectory on the central computer (22) and on the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) holds data about customer orders placed on the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).
- 5 16. The system as claimed in claim 14 wherein a second subdirectory on the central computer (22) and on the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) holds data about order process of customer orders placed on the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).
- 10 17. The system as claimed in claim 14 wherein a third subdirectory on the central computer (22) and on the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) holds files and data, which are used to carry out the update of the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>).
18. The system as claimed in any of the claims 10 to 17 wherein the external station (20<sub>1</sub>, 20<sub>2</sub>, 20<sub>3</sub>, ...20<sub>n</sub>) is an order station for ordering prints of digital photographic data gained from various sources.

15

20

25

30

35

40

45

50

55

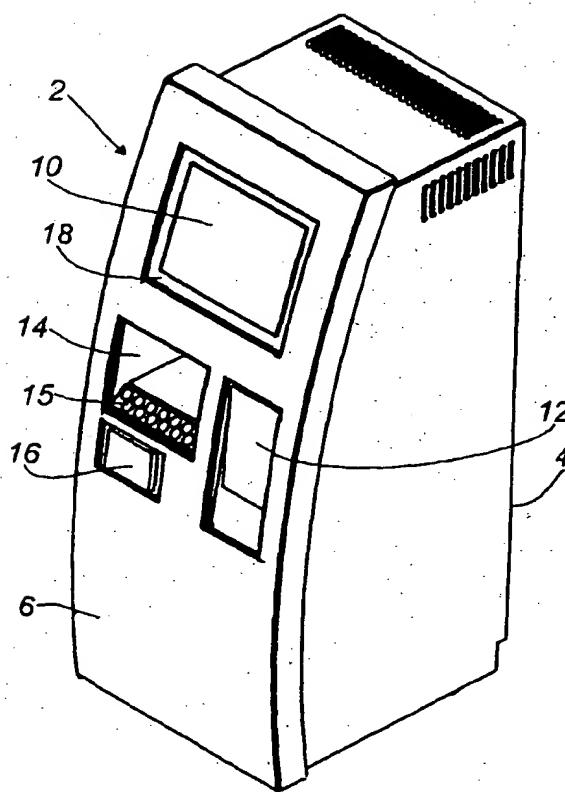


Fig. 1

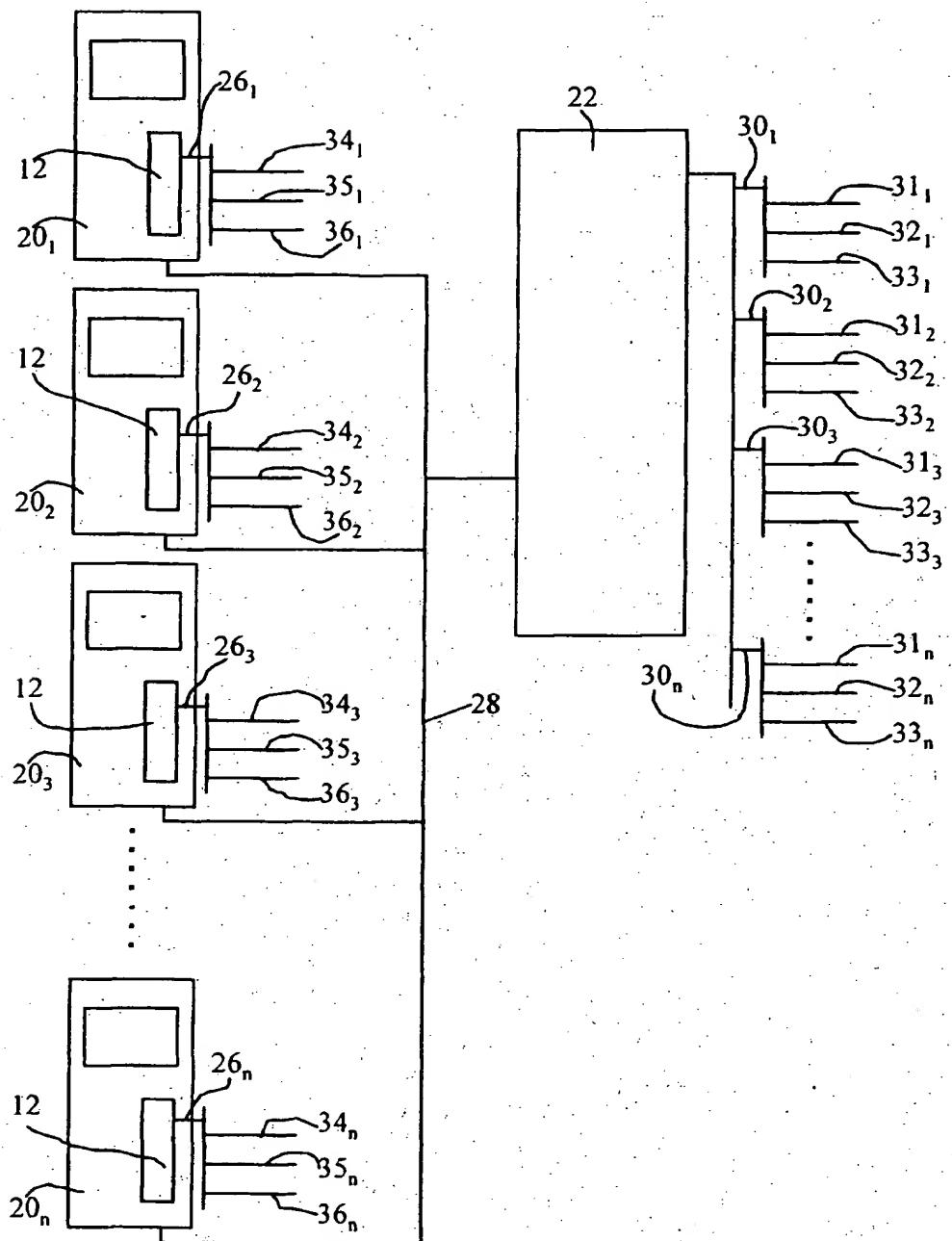


Fig. 2

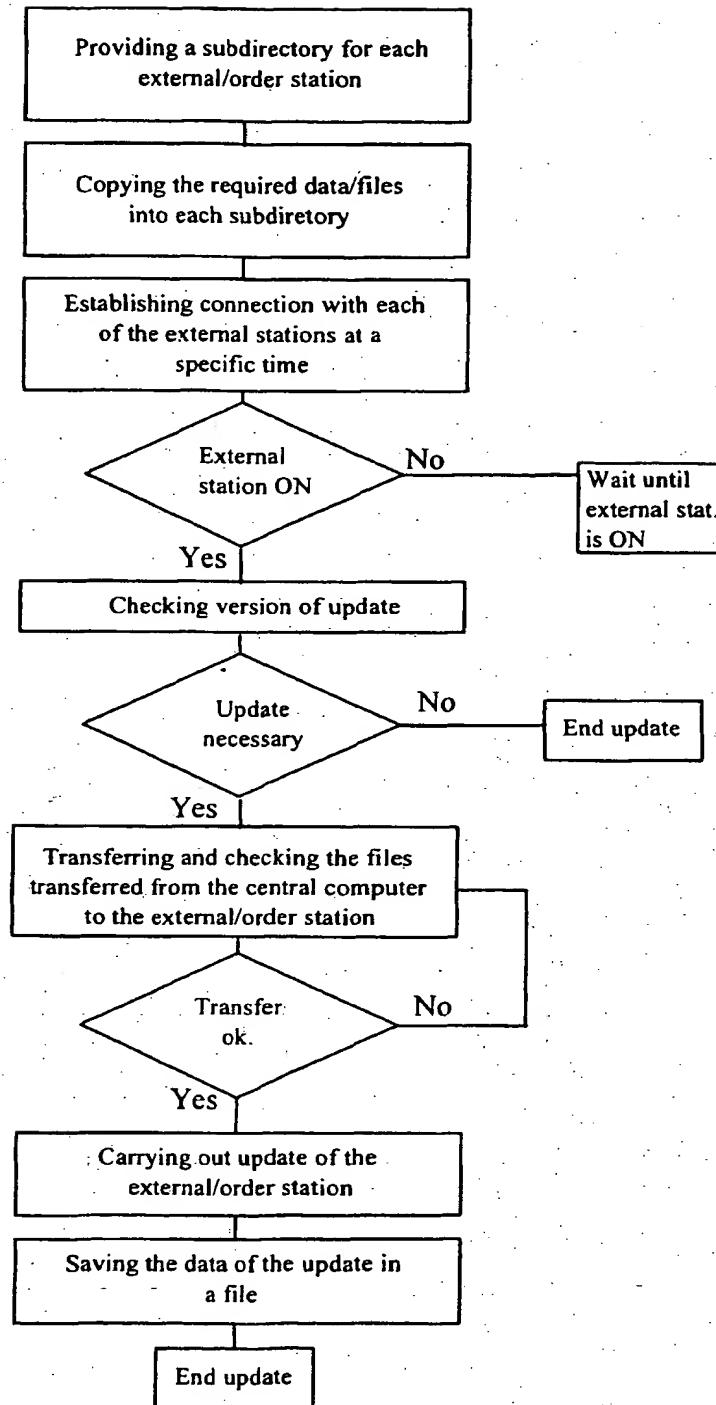


Fig. 3

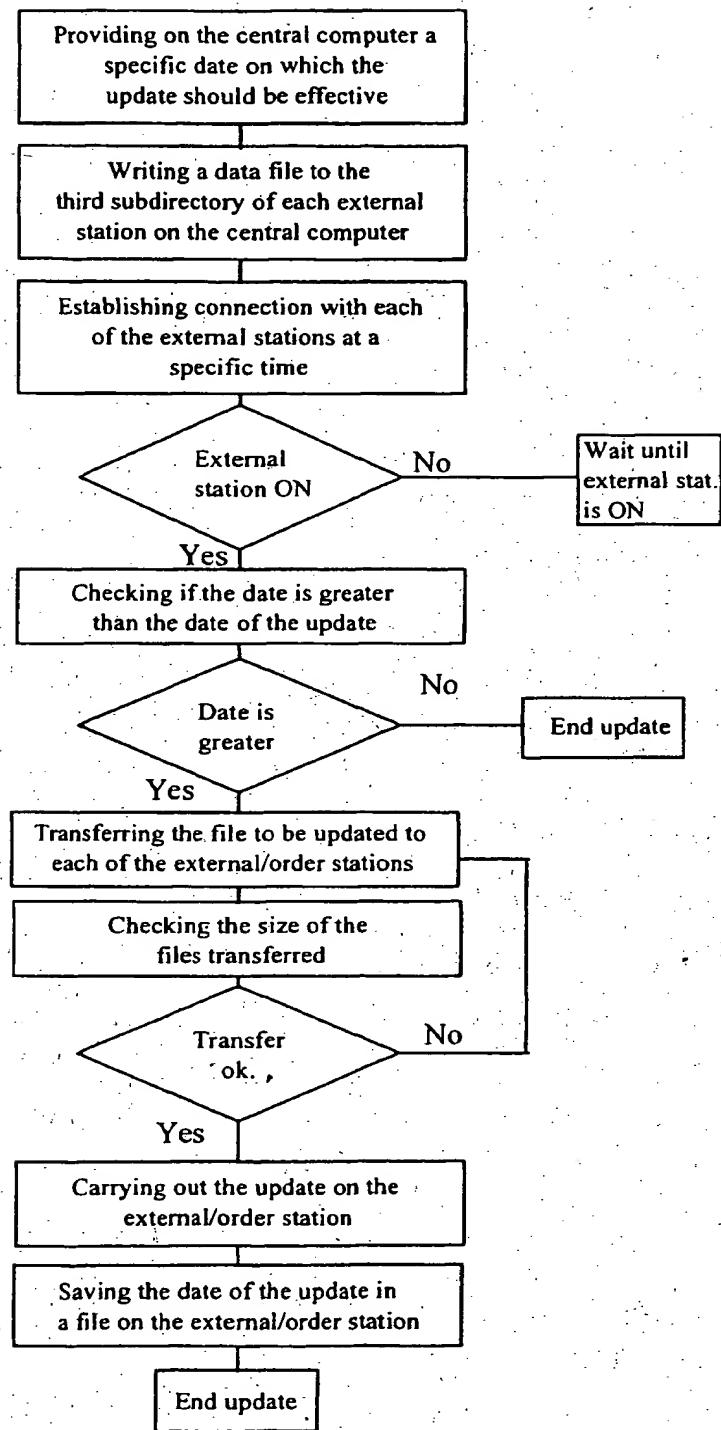


Fig. 4

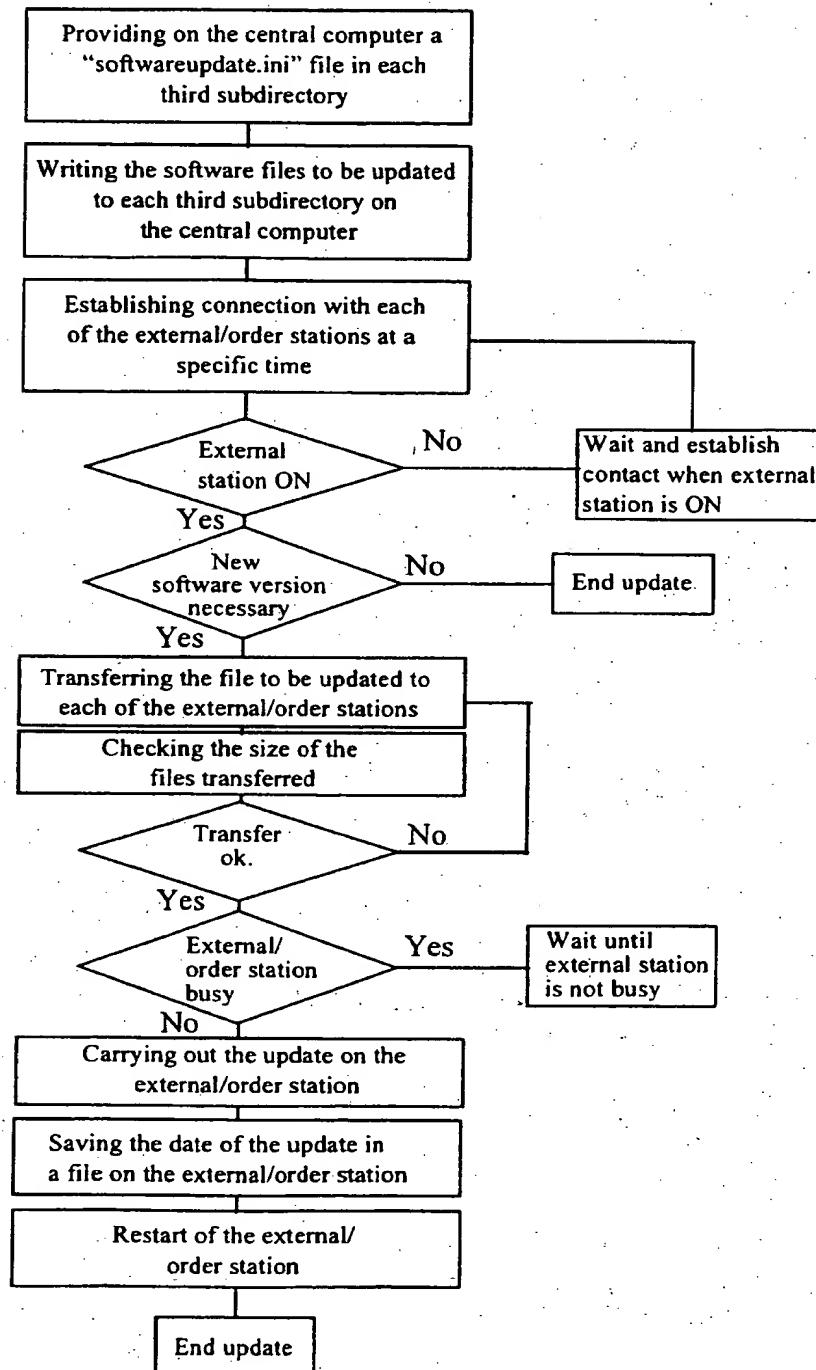


Fig. 5

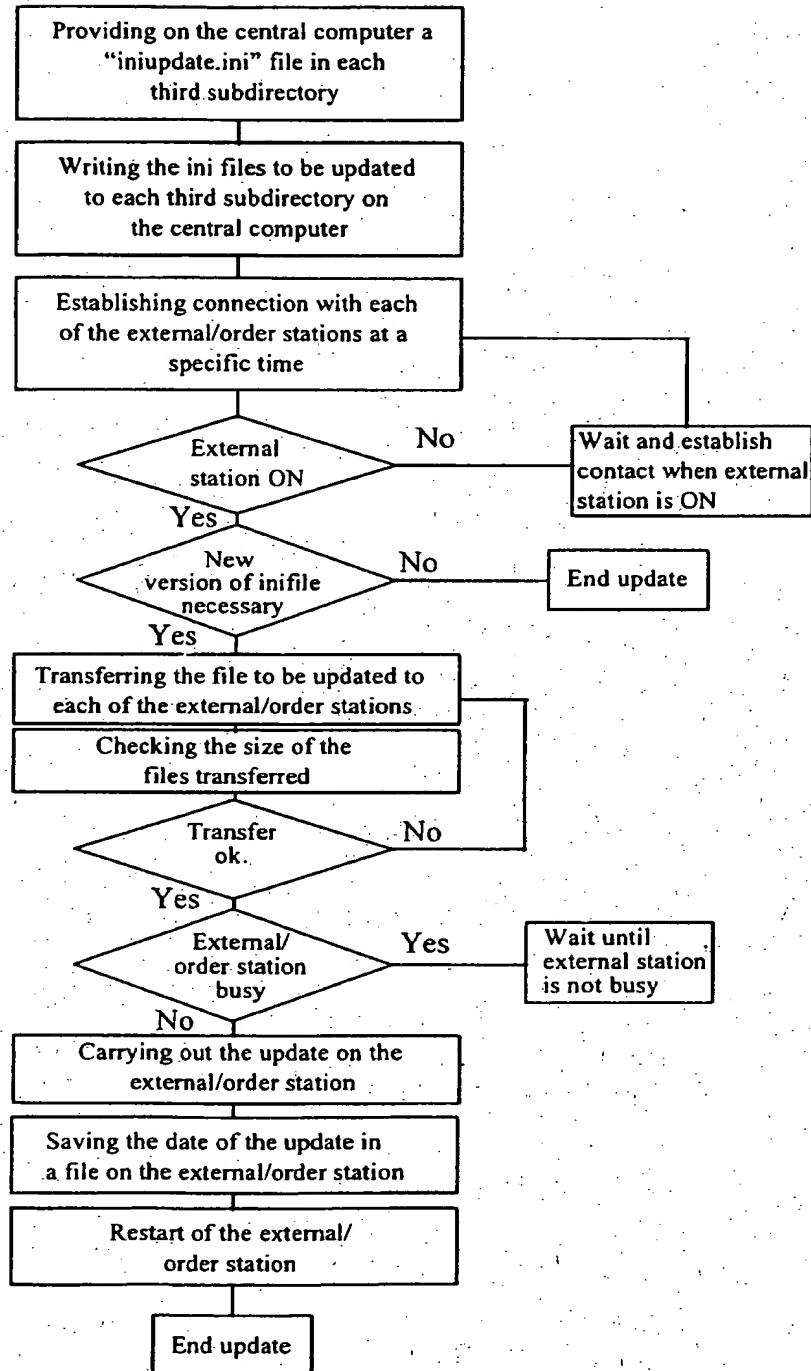


Fig. 6



European Patent  
Office

## EUROPEAN SEARCH REPORT

Application Number  
EP 98 11 7504

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
A	US 5 421 009 A (PLATT STEPHEN M) 30 May 1995 * column 1, line 58 - column 2, line 42 *	1-17	G06F9/445 G06F9/44
A	WO 94 25913 A (NOVADIGM INC) 10 November 1994 * page 14, line 10 - page 16, line 9 *	1-17	
A	"LOCAL AREA NETWORK PROGRAM DISTRIBUTION FACILITY" IBM TECHNICAL DISCLOSURE BULLETIN, vol. 33, no. 5, 1 October 1990, pages 169-171, XP000107424 * the whole document *	1-17	
A	WO 97 31312 A (CITRIX SYSTEMS INC) 28 August 1997 * page 3, line 23 - page 5, line 23 *	1-17	
TECHNICAL FIELDS SEARCHED (Int.Cl.6)			
G06F			
The present search report has been drawn up for all claims			
Place of search	Date of completion of the search		Examiner
THE HAGUE	11 March 1999		Brandt, J
CATEGORY OF CITED DOCUMENTS			
X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document			
T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

**CLAIMS INCURRING FEES**

The present European patent application comprised at the time of filing more than ten claims.

Only part of the claims have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims and for those claims for which claims fees have been paid, namely claim(s):

1-17

No claims fees have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims.

**LACK OF UNITY OF INVENTION**

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

All further search fees have been paid within the fixed time limit. The present European search report has been drawn up for all claims.

Only part of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the inventions in respect of which search fees have been paid, namely claims:

None of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the invention first mentioned in the claims, namely claims:

ANNEX TO THE EUROPEAN SEARCH REPORT  
ON EUROPEAN PATENT APPLICATION NO.

EP 98 11 7504

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

11-03-1999

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5421009	A	30-05-1995	NONE		
WO 9425913	A	10-11-1994	AT	156282 T	15-08-1997
			CA	2160907 A	10-11-1994
			DE	69404647 D	04-09-1997
			DE	69404647 T	27-11-1997
			EP	0696368 A	14-02-1996
			JP	8510341 T	29-10-1996
			WO	9425923 A	10-11-1994
			WO	9425924 A	10-11-1994
			US	5581764 A	03-12-1996
WO 9731312	A	28-08-1997	AU	2272797 A	10-09-1997
			EP	0882263 A	09-12-1998
			NO	983858 A	21-10-1998

This Page Blank (uspto)